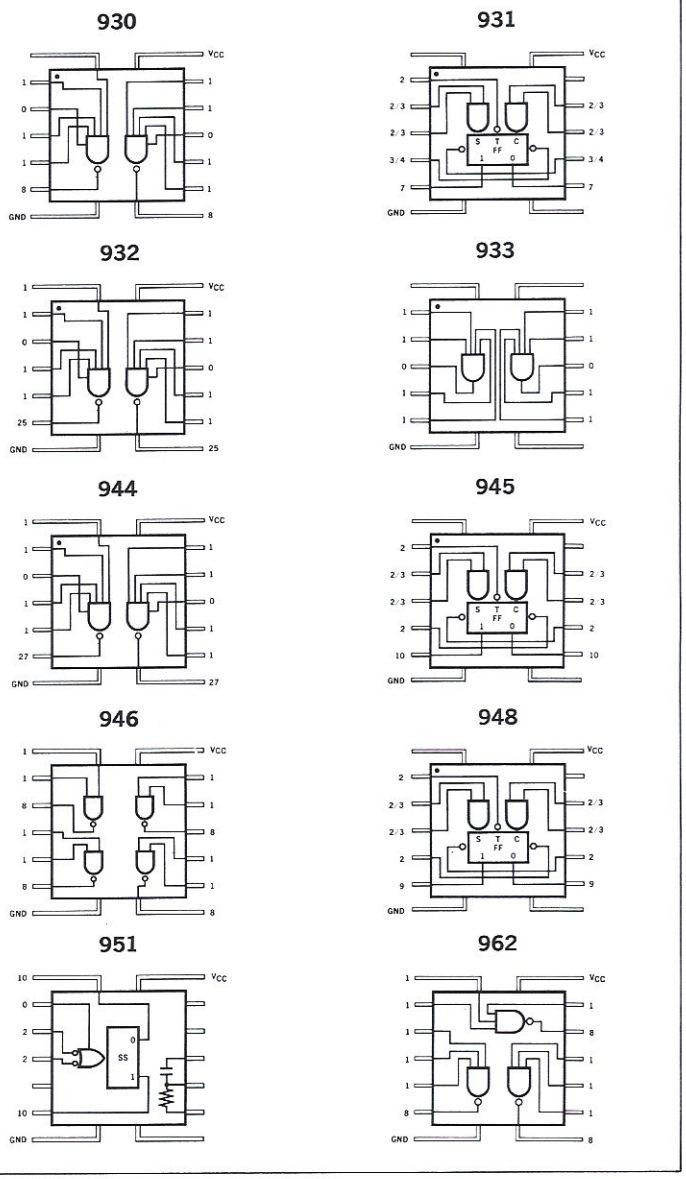
DIODE-TRANSISTOR MICROLOGIC® INTEGRATED CIRCUITS

by George Powers

FIG. 1. Diode-Transistor Micrologic® IC Family.



I. INTRODUCTION

Fairchild Diode-Transistor Micrologic® DTμL integrated circuits, is a family of compatible integrated logic building blocks using diode-transistor logic and designed specifically for integrated-circuit technology. The advantages of this family, along with distinctly superior performance, are as follows:

- Low power dissipation
- Operation over a wide range of supply voltages
- High performance with a single power supply over supply voltage range
- High worst-case noise immunity
- High fan-out capability
- Fan-out and noise immunity trade-off
- Specified operation over entire military temperature range

II. DESCRIPTION OF ELEMENTS

Each element of the DTμL family of integrated circuits and its function is listed in Table I.

All of the DTμL elements are available in 14-lead flat packages or in TO-5 style headers. As only 10 leads are available on the TO-5 header, some input functions must be omitted. A selection of these units is also available in the new 14-lead dual in-line package for commercial, industrial, and limited temperature applications. Figure 1 indicates the logic configuration of each of the elements, together with input and output loading rules.

DTμL Gate

The basic circuit of the DTμL family is shown in Fig. 2. The salient feature of this circuit is the replacement of one of the offset diodes usually found in conventional circuits by a transistor whose

COPYRIGHT FAIRCHILD SEMICONDUCTOR, 1966 • PRINTED IN U.S.A. 2320-267-86 10M

313 FAIRCHILD DRIVE, MOUNTAIN VIEW, CALIFORNIA, (415) 962-5011, TWX: 910-379-6435

FAIRCHILD SEMICONDUCTOR
A DIVISION OF FAIRCHILD CAMERA AND INSTRUMENT CORPORATION

Fairchild cannot assume responsibility for use of any circuitry described other than circuitry entirely embodied in a Fairchild product. No other circuit patent licenses are implied.

collector is returned to the midpoint of the input pull-up resistor. By this means the relative gating efficiency is double that obtained with conventional DTL design; hence, the h_{FE} requirement of the output transistor is reduced by one-half for a given fan-out or, conversely, the fan-out capability is doubled for a given h_{FE} requirement. The compromises usually necessary in power dissipation, component tolerance, and fan-out capability are thus greatly relaxed, permitting performance unattainable with the conventional approach. A comprehensive description of DTμL circuitry and manufacturing technique is contained in "Diode-Transistor Micrologic for Digital Applications," by R. E. Bohn and R. B. Seeds, appearing in the August 3, 1964 issue of Electronic Design.

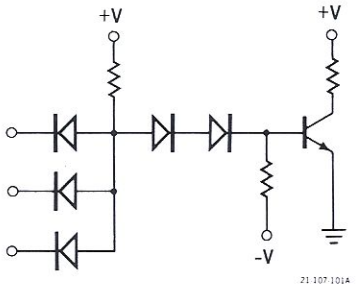


FIG. 2a. Conventional DTL.

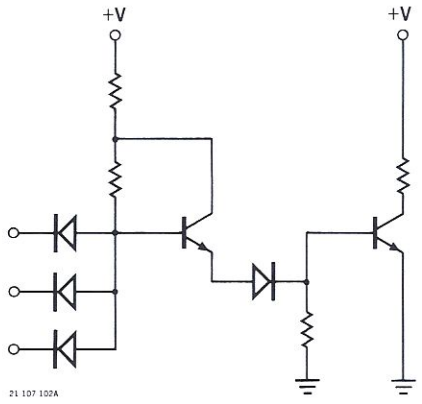


FIG. 2b. Diode-Transistor Micrologic® IC.

Logic Function. Logically speaking, the Gate is either a NAND or a NOR gate, depending upon the voltage level selected for the logical "1" condition. This dual relationship is described in the data sheet and shown with logic symbols in the format of MIL-STD-806B, Graphic Symbols for Logic Diagrams.

The positive-logic form is the NAND (AND-invert) function. The shape of the symbol indicates the AND function, and the inverting property is portrayed by the change of active (logical "1") signal level from input to output. For all of the element symbols, a small circle at an input or output terminal signifies that the active level is the less positive of the two logic voltages. If the circle is omitted, the active level is the more positive of the two logic voltages. The negative-logic form is the NOR function. The shape of the symbol indicates the OR function, and, as in the case of the NAND function, inversion is indicated by a change of active signal level from input to output, but in the opposite sense.

Three principal variations of the Gate are available: The DTμL 930 Dual 4-Input Gate, the DTμL 946 Quad 2-Input Gate, and the DTμL 962 Triple 3-Input Gate. The 930 Gate is provided with input extension terminals for use when fan-in greater than 4 is needed. The DTμL 933 Dual 4-Input Extender may serve this function, or silicon diodes such as the 1N914 or any of the Fairchild FD series may be substituted.

Wired OR Function. The outputs of two or more DTμL Gates may be paralleled directly to produce a Wired OR function. This connection is very useful and can

TABLE I

ELEMENT	FUNCTION
DTμL 930, Dual 4-Input Gate	Low-level NAND/NOR Gate
DTμL 946, Quad 2-Input Gate	Low-level NAND/NOR Gate
DTμL 962, Triple 3-Input Gate	Low-level NAND/NOR Gate
DTμL 932, Dual 4-Input Buffer	High-drive Gate
DTμL 944, Power Gate	High-drive Gate
DTμL 933, Dual 4-Input Extender	Fan-in Expander
DTμL 931, Clocked Flip-Flop	All-purpose Storage
DTμL 945, Clocked Flip-Flop	All-purpose Storage
DTμL 948, Clocked Flip-Flop	All-purpose Storage
DTμL 951, Monostable Multivibrator	Pulse Generation

often result in a reduction in the total amount of circuitry to implement a function. Two examples which exploit the Wired OR function are given in the applications section (see Figs. 19 and 20). The Wired OR is symbolized in these figures by enclosing the output connection point with a small logic symbol of appropriate shape. Since both of the examples use positive logic, the Wired OR node is enclosed with an AND symbol.

Individually, the Gates in the Wired OR are NAND Gates for positive logic. If one of two such Gates has inputs A and B and the other has inputs C and D, their outputs are $\overline{A \cdot B}$ and $\overline{C \cdot D}$, respectively. From the circuit diagrams of the Gate, it is clear that either output may pull the connection point to ground, but both must be off for the output to rise to the high level. Since this latter condition is the one desired, and the statement that both must be off is precisely the AND function, the AND symbol is chosen. However, by DeMorgan's theorem, the following expression holds (hence, the name Wired OR):

$$\bar{A}\bar{B} \cdot \bar{C}\bar{D} = \overline{AB + CD}$$

Wired OR connections impose loading on outputs just as inputs do. For the sake of simplicity, each added output up to 5 is considered equivalent to a unit of fan-out loading. Since the actual loading is less, six added Gates represent only five units of fan-out.

DTμL Buffer and Power Gate

When high fan-out or unusual capacitive loading is encountered, the DTμL 932 Dual 4-Input Buffer or 944 Power Gate may be used to advantage (see Fig. 3).

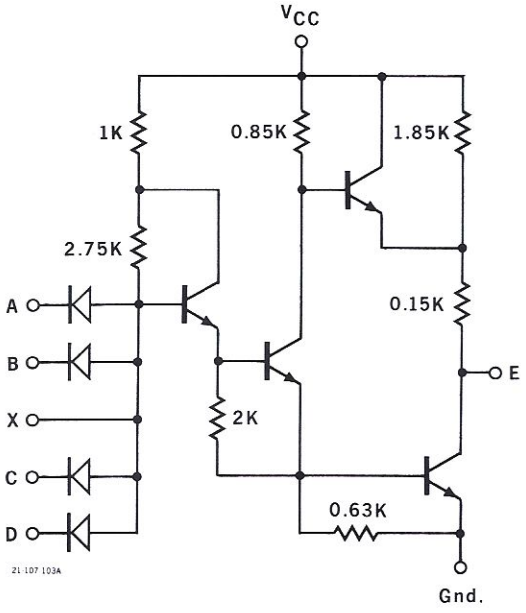


FIG. 3a. Schematic Diagram of the DTμL 932 Buffer.

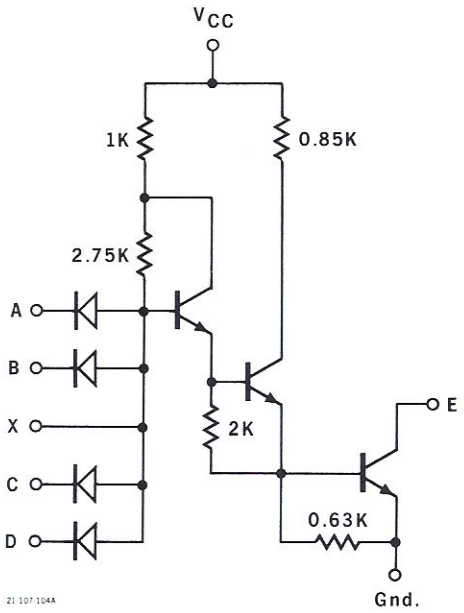


FIG. 3b. Schematic Diagram of the DTμL 944 Power Gate.

The DTμL Buffer is ideally suited to driving heavy capacitive loads because its output presents a very low impedance when turned off as well as when turned on. Resulting RC time constants are very short, thereby improving system operation.

The cascade output, which achieves low driving impedance, precludes connecting outputs to obtain the Wired OR function. In the DTμL Power Gate, the output emitter follower of the DTμL Buffer is omitted while retaining the high-current, grounded-emitter output. While the turn-off speed of the Power Gate is not on a par with the Buffer, the fan-out capability is actually greater, and the ability to Wire OR is recovered. Furthermore, the uncommitted output terminal can be used very effectively at system interfaces.

DTμL Clocked Flip-Flop

The Clocked Flip-Flop is two simple latch circuits composed of cross-coupled DTμL Gates (see Fig. 4). These simple latches are connected in tandem through clock-coupling transistors (Q_7 and Q_8) whose function is to isolate the second latch ("slave") from the first latch ("master") when the latter is permitted to change. The "master" is enabled through input AND gates by the same input signal that inhibits the clock-coupling transistors and thus isolates the "slave."

Excluding the connection of the clock-coupling transistors Q_7 and Q_8, the "slave" is simply a cross-connected pair of 2-Input DTμL Gates. As noted in the preceding paragraphs, the negative-logic interpretation of the Gate is the OR function with inversion.

Thus, if either diode input is low, the corresponding output must rise to the high level. Without external influence, an equilibrium will be established in which the low output of one Gate causes the other Gate output to be high, in turn holding the first output low.

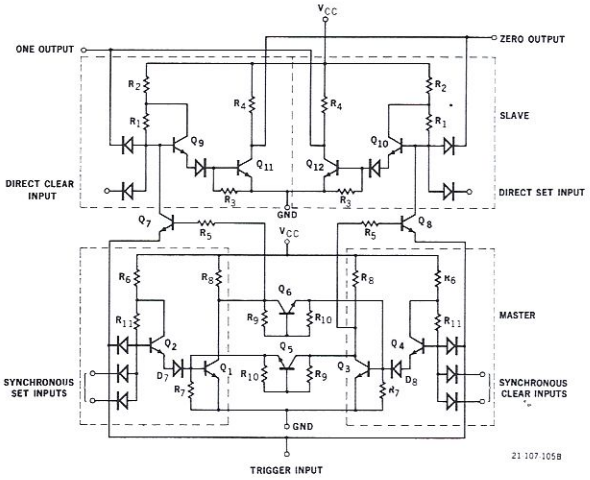


FIG. 4. Schematic Diagram of DTμL 931 Clocked Flip-Flop.

The direct or asynchronous inputs are connected to these OR Gates, and the application of a low signal to either input will cause the corresponding output to rise to the high level. In the event that both direct inputs are simultaneously low, both outputs will be concurrently high. The last input being low will predominate (see Fig. 5).

The clock-coupling transistors may also inject a low level into the "slave" and therefore act as OR inputs. However, the conditions necessary for a low level at the collector of a clock-coupling transistor are a low level at the emitter and a high level at the base.

The cross-connection of DTμL Gates in the "master" is somewhat more complex. A special network composed of Q_5, Q_6, and the resistors labeled R_9 and R_{10}, along with the diodes D_7 and D_8, form the mutual OR connections necessary for realizing the latching action desired. In this case, however, the OR is for positive levels, and consequently the input diodes provide a positive AND function.

Three diodes are provided in each input AND gate. Two of these are intended as synchronous logic inputs. The third diode of each AND gate is connected to the Trigger input and regulates the entry of synchronous information to the "master."

The Trigger input also extends to the emitters of the clock-coupling transistors. It is apparent that a high level applied to the Trigger input will inhibit the clock-coupling transistors while enabling the input AND gates. Thus, the "slave" is indeed isolated when the "master" is permitted to change. A low level applied to the Trigger input will have just the opposite effect; that is, the input AND gates will be inhibited, precluding the other AND gate inputs from exerting any influence on the state of the "master," while the "slave" is directly connected through the enabled clock-coupling transistors.

ASYNCHRONOUS ENTRY*

DIRECT INPUTS		OUTPUTS	
SET	CLEAR	ONE	ZERO
H	H	X	$\overline{X}$
H	L	L	H
L	H	H	L
L	L	H	H

*Trigger input of DTμL 931 must be high during asynchronous entry.

H = high

L = low

X = immaterial - either high or low with equal effect.

SYNCHRONOUS ENTRY

SYNCHRONOUS INPUTS AT t = n				OUTPUTS AT t = n + 1	
SET_1	SET_2	$CLEAR_1$	$CLEAR_2$	ONE	ZERO
L	X	L	X	X	$\overline{X}$
L	X	X	L	X	$\overline{X}$
X	L	L	X	X	$\overline{X}$
X	L	X	L	X	$\overline{X}$
L	X	H	H	L	H
X	L	H	H	L	H
H	H	L	X	H	L
H	H	X	L	H	L
H	H	H	H	AMBIGUOUS	

21-107-106A

FIG. 5. Clocked Flip-Flop Logic Characteristics.

This "escapement" action of the "master" and "slave" working together assures reliable operation as a Clocked Flip-Flop independent of trigger waveshape or speed. Rise and fall times are irrelevant — a sine wave, or any other waveshape having distinct low and high levels, is a reliable trigger signal.

To guarantee the escapement action, it would be adequate if the enabling threshold of the input AND gates were coincident with the inhibiting threshold of the clock-coupling transistors. Obviously, the maintenance of such a match would be difficult even if the tolerance of circuit components could be closely controlled. Fortunately, it is very easy to separate the thresholds and thereby guarantee reliable operation.

The logic properties of the DTμL Clocked Flip-Flop, including active logic levels, are summarized by its logic symbol (see Fig. 1). The reason for the active levels shown on the symbol is apparent from the circuit diagram, but a question might arise as to the choice of polarity for the Trigger input, and the reason for separating it from the input AND gates.

The polarity selected for the trigger active level is intended to indicate the slope of the trigger signal which actually causes the outputs of the device to change. This transition (to the low level) is a reference point from which transfer characteristics may be measured and specified. Obviously, the opposite transition (to the high level) has no apparent external effect. What is more important to the logic designer, the negative transition of the trigger signal marks the time from which all logic delays must be measured. To maintain well-defined operation, all logic signals must be stable at the rise of the trigger signal. Therefore, logic propagation time is the period between the falling trigger and the next rising trigger signal. To achieve the greatest time efficiency, the positive dwell of the trigger signal should be minimized (no less than 50 nsec) so that the largest portion of a clock period can be used for signal propagation.

The trigger has been separated from the input AND gates for the following reasons:

- To simplify, at least in part, a very complex symbol.
- To consolidate the Set, Trigger, and Clear designations by placing the trigger at the face of the flip-flop box, and to emphasize the synchronous nature of the inputs in a manner that would be unclear otherwise. In this way, the AND function between the inputs S_1 and S_2 and C_1 and C_2 is clearly brought into focus.

The truth tables of Fig. 5 summarize the responses of the Clocked Flip-Flop to all of the possible input conditions. Relative voltage levels — H = high, L = low — are used for clarity. Where the actual level is immaterial, an X appears.

The Trigger input does not appear in the synchronous entry table, but its influence is implicit. Inputs are stated at time t = n, and outputs are stated at time t = n + 1. The demarcation between times n and n + 1 is an excursion of the Trigger input from low to high and back to low again. This unit time delay due to clocking is typical of synchronous operation.

In response to signals applied to the direct inputs, both the Clocked Flip-Flop outputs may be concurrently high. However, the outputs are always complementary in response to synchronous entry. Each pair of synchronous inputs must be high to affect the output. In the event that all synchronous inputs are high simultaneously, an unpredictable output condition will result. This ambiguity is characteristic of R-S Flip-Flops and must be avoided. The twin Set and Clear inputs of DTμL Clocked Flip-Flops simplify the resolution of R-S characteristic ambiguity.

Since the Clocked Flip-Flop outputs are always complementary in response to synchronous entry, they may be cross-connected to one of each of the Set and Clear inputs to assure that the ambiguous input condition can never occur. The terminal characteristics of a Clocked Flip-Flop so connected are those of a J-K Flip-Flop. The term J-K is applied to a flip-flop which toggles whenever its inputs are both simultaneously active at clock time. This mode of operation is particularly valuable in the construction of counters as will be explained in the applications section of this note.

The crossover in the feedback path is essential or the flip-flop would be able to change only to the state in which it currently resides — a useless situation.

The DTμL Clocked Flip-Flop exists in three forms: the DTμL 931, the DTμL 945, and the DTμL 948. The DTμL 945 and 948 differ from the DTμL 931 in two respects; they have output buffer circuits, and direct inputs affect the "master" latch as well as the "slave."

Output buffers isolate the "slave" from the output load and increase the fan-out capability. Output turn-on and turn-off times are more nearly matched in the 945 and 948 than they are in the 931, but are also slightly longer. Wired ORing becomes a possibility in the 945 and 948, but examples of its usage are difficult to conceive.

The effect of direct entry to both "master" and "slave" in the DTμL 945 and 948 is to eliminate the need of holding the trigger input high during asynchronous entry. This requirement is necessary in the 931 to prevent a possible conflicting state of the "master" from affecting the "slave" during asynchronous entry. Since "master" and "slave" must agree in the 945 and 948, the restriction is unnecessary. The prime example of this type is arbitrarily preset ripple-carry counters. The DTμL 945 and 948 have no restriction and function well in this capacity. The DTμL 948 has smaller output resistors than the 945, 2 K ohms rather than 6 Kohms, for higher speed operation.

DTμL Monostable Multivibrator

Figure 6 shows the schematic of the DTμL 951 Monostable Multivibrator. The 951 has two complementary outputs and an extendable two-diode OR input. The DTμL 933 Dual 4-Input Extender or silicon diodes such as the Fairchild FD series, the 1N903, and the 1N914 are all useful for extending fan-in. The 1N914 is available to MIL-S-19500/116 (USN), if desired.

The DTμL 951 is triggered by a falling voltage transition of at least 1 volt. The transition rate should exceed 40 mV/nsec. Therefore, a 1.0-volt transition should not exceed 25 nsec, while a 4.0-volt transition may take as long as 100 nsec.

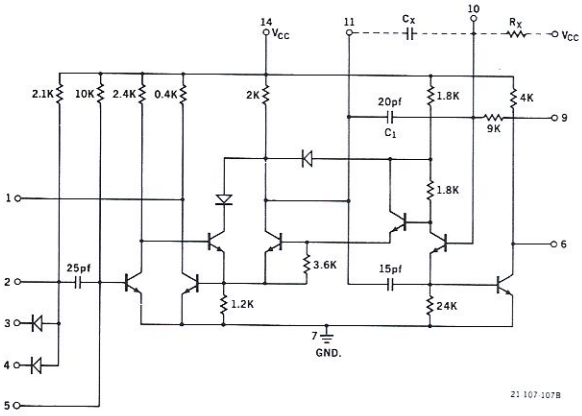


FIG. 6. Schematic Diagram of DTμL 951 Monostable Multivibrator.

The inherent pulse width of the DTμL 951 is approximately 100 nsec. Pulse width can be modified by attaching external components to pins 9, 10, and 11. A diffused resistor of 9 Kohms $\pm$15% is connected between pins 9 and 10. Whenever an external resistor is not used, pin 9 must be connected to the supply voltage.

External capacitance may be added between pins 10 and 11 to increase the output pulse width. Using the diffused 9 K-ohm resistor, the output pulse width may be calculated from the following expression:

$$T = 4.5\ (C_x + 20)$$

where T is in nsec, and C is in picofarads. This equation is subject to the tolerance of the timing components.

Pulse-width accuracy may be improved by the use of a precision resistor and capacitor to one percent or less over the full temperature range of -55°C to +125°C. The external resistor may range between 9 Kohms and 15 Kohms and is connected between pin 10 and the supply voltage (pin 9 is not used).

Under these circumstances, pulse width may be calculated from the following:

$$T = 0.5\ R_x\ (C_x + 20)$$

where T is in nsec, R_x is in Kohms, and C_x is in pf.

Expressions given for calculating pulse width assume a duty cycle not exceeding 40 percent. This limitation is imposed by the capacitor recharge time constant. The duty cycle may be increased to 50 percent by connecting a 2 K-ohm resistor between pin 11 and the supply voltage. For duty cycles greater than 50 percent, the designer should consider using the complementary output. Output pin 6, normally at the low level, rises to the high level when triggered. Output pin 1, normally at the high level, falls to the low level when triggered. The transition times of either output are independent of pulse width. Rise times are subject to loading, but fall times are typically 10 nsec. Thus, one DTμL 951 may drive another directly.

Minimum pulse width at pin 1 is controlled by the RC time constant at the input. Should a pulse narrower than 100 nsec be required from pin 1, the time constant may be reduced by connecting a resistor between pin 5 and the supply voltage. A 10 K-ohm resistor will cut the minimum pulse width approximately in half. The possibility of false triggering due to narrow voltage spikes may be reduced by attaching a capacitor from pin 5 to ground. This measure is subject to several factors and should be approached experimentally when the need arises.

III. APPLICATIONS OF DTμL ELEMENTS

The application about which more design questions are asked than any other is the counter. Since the logic flexibility of DTμL is attested by the simplicity of counters, this topic is dwelt upon at some length in the discussion that follows. Four separate decade counters using only Clocked Flip-Flops (no external gates whatever) illustrate the versatility of DTμL. The example in Fig. 19, a reversible Mod-10 Shift Counter,

shows the Wired OR function used to minimize an otherwise bulky configuration; note that a total of only nine integrated circuit packages are used.

Ripple-Carry Counters

The simplest method of counting employs an asynchronous technique often described as a ripple carry. An example of a 4-stage ripple-carry counter is given in Fig. 7.

Each stage in any binary counter has its outputs cross-connected to its inputs so that count pulses are routed alternately to the two inputs. Each output waveform, therefore, is a square wave at one-half the frequency of its input. By using the output of one stage as the input to the next, the frequency is successively halved.

Obviously, either output could be connected to the next stage, but by using the connection shown in Fig. 7, the waveforms can be considered as weighted binary values. Thus, the first stage has a value of 1, the second a value of 2, the third a value of 4, and so on. By adding the values of all stages in the "1" state, total count is obtained. In this way, these simple frequency dividers act as a binary counter.

Observe that when any stage in a ripple-carry counter is to change, all preceding stages must also be changing from "1's" to "0's." This situation is analogous to adding 1 to an all-1 binary number. A 1 added to the least significant bit causes that bit to become a 0 with a carry to the next bit. Each bit in turn becomes a 0 with a carry to the next bit, and so on. The carry propagates bit by bit until it finally encounters a 0 and there ends.

Similarly, in the ripple-carry counter, an input causes the least significant stage to change to "0," which in turn causes the next stage to change to "0," and so on until it terminates at a stage which changes from "0" to "1." A finite amount of time is consumed as each stage changes sequentially, and the carry seems to ripple through the counter (hence, the name).

When used for frequency division, a ripple-carry system has no ill effects because the only thing of importance is the frequency of the output at the final stage.

But when it is necessary to use the specific binary-coded outputs of two or more stages as inputs to a logic gate, the time difference between changing outputs can cause errors. As the individual delays are cumulative, this situation is aggravated when the outputs are from stages widely separated in the counter. In fact, unless special measures are taken, delays can make ripple counters unsuitable in many applications.

Flip-Flop Logic Flexibility

Many integrated circuit flip-flops have only one clocked Set and one clocked Clear input and these must be cross-connected to their own outputs for counting. Such flip-flops are, of course, suited only to ripple counting techniques, and all of their attendant difficulties. J-K Flip-Flops, such as the μL916 J-K Flip-Flop, have internal output-to-input cross-connections, and consequently are not restricted to ripple-carry counting. However, internal cross-connections are not always needed. Therefore, in the DTμL Clocked Flip-Flops, the use of all clocked inputs is left to the discretion of the designer.

As indicated by the logic symbol, the two clocked Set inputs and the two clocked Clear inputs are combined in AND gates at the flip-flop inputs. While the J-K mode of operation can still be attained by cross-connecting the outputs to the inputs, the designer is free to use the synchronous inputs in a manner that is most advantageous to the situation at hand.

Synchronous Counters

The objectionable cumulative delay of ripple-carry counters can be effectively avoided through the use of synchronous techniques. Synchronous counters employ a Clock input signal which is common to all stages. The clock signal triggers all stages simultaneously (see Fig. 8) and all outputs which are scheduled to change do so together. In fact, the only differences present are due to small inherent variations of the response time of the individual stages which in practice are negligible. Hence, it is possible to decode specific counts without concern for unwanted gating spikes or shortened pulses.

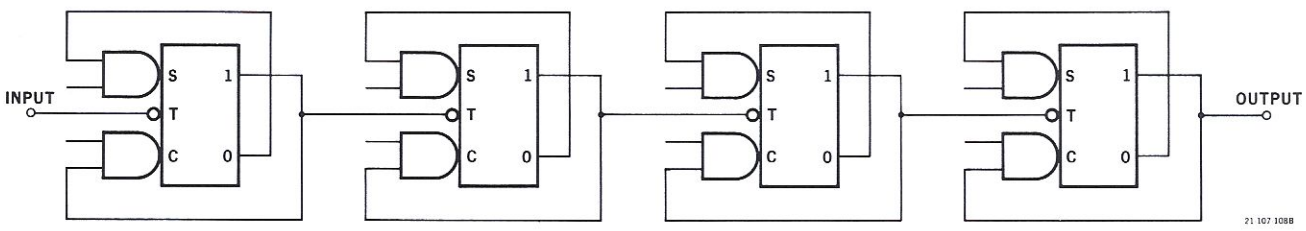


FIG. 7. Ripple-Carry Binary Counter.

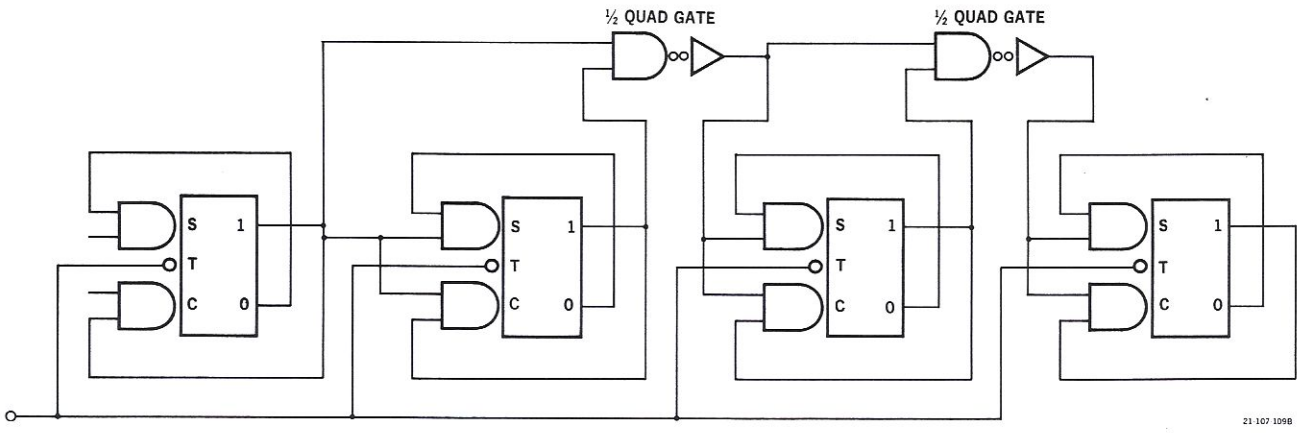


FIG. 8. Synchronous Serial-Carry Binary Counter.

The synchronous counter of Fig. 8 employs serial-carry logic. A carry gate is needed ahead of each stage except the first two. The first stage of a binary counter changes with each Clock input pulse. Therefore, only J-K cross-connections are needed. The remaining synchronous logic inputs are left open. The second stage input is conditional; a "1" in the first stage is required. Thus, while the first two stages have the same Trigger input frequency, the second stage can respond only to each second clock input; both the Set and Clear inputs are inhibited when the first stage output is "0."

At the input to the third stage, an AND gate is needed to combine the "1" outputs of the first two stages. An inverter is also used to establish the proper polarity for the inputs of the third stage. The output of this carry gate will enable the synchronous logic inputs of the third stage for each fourth Clock input. Consequently, the third stage outputs will change at one-eighth the clock rate. In addition, these outputs will always be phased with preceding stages so that binary weighting may be assigned to them.

The carry gate to each subsequent stage combines the output of the previous carry gate with the "1" output of the previous flip-flop. Thus, a carry generated at the first stage will be propagated by each consecutive stage whose state is "1" and terminated at the first stage whose state is "0."

It should be apparent that the chain of carry gates introduces an overall speed limitation because of the serial-carry propagation time. Sufficient time must be allocated between successive clock pulses for a carry to traverse the entire carry chain. This time is the sum of the delays introduced by each carry gate, and is in the range of 20 to 25 nsec per element for the typical DTμL NAND/NOR gate at room temperature.

Carry propagation time may be accelerated by a parallel arrangement of the carry gates (see Fig. 9). This arrangement differs in that the "1" output of all less significant stages goes directly to the input of each carry gate. The total carry propagation time is now simply that of one carry gate, whereas in the serial carry arrangement it is the sum of the propagation times of all carry gates.

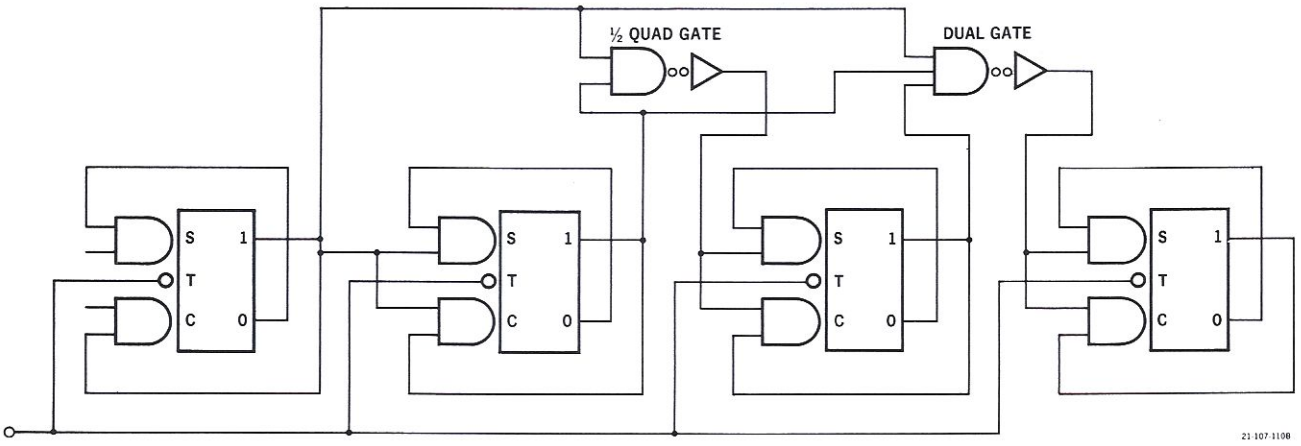


Fig. 9. Synchronous Parallel-Carry Binary Counter.

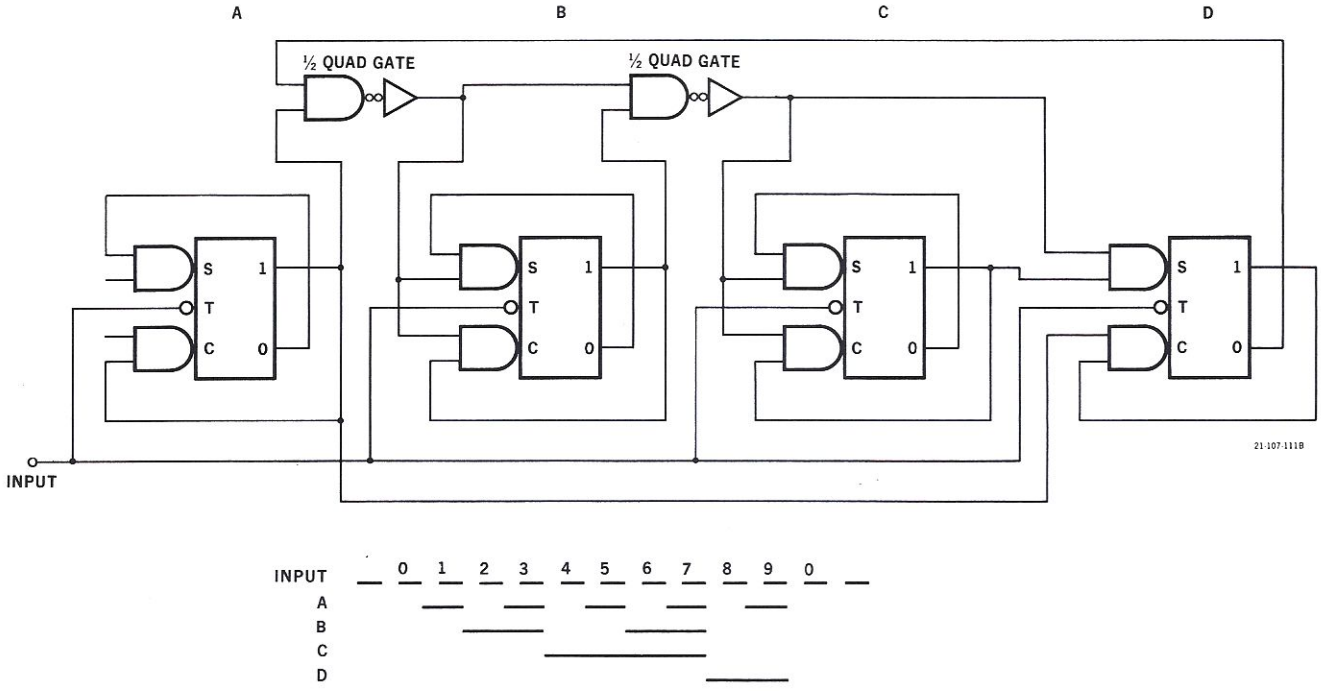


FIG. 10. Synchronous Serial-Carry 8-4-2-1 Decade.

The fan-in requirement of each successive gate in a parallel-carry counter increases by one. Carry gate fan-in may be easily extended up to 20 with the DTμL Input Extender element without degrading the performance of the unit beyond the effects of capacitance on the extension wiring. The fan-out capability of the less significant stages will be consumed before this, as each additional carry gate imposes one additional load on each less significant stage "1" output. In large parallel counters, the DTμL Buffer Driver element may be needed to provide sufficient drive to the entire carry system.

A compromise between speed and cost in large counters can be reached by arranging stages in groups. Intra-group carry may then be parallel without exceeding fan-out capability while inter-group carry is serial. The size of groups may be adjusted to the particular situation.

Flip-flops in which all synchronous inputs must be used as feedback connections require from two to three times more gates in the carry logic in synchronous counters than the examples just cited — a dramatic advantage of the multiple inputs of the DTμL Clocked Flip-Flop.

Decade Counters

Both the ripple-carry counter and the synchronous counter can be modified to count in a closed sequence of 10. In either case, a binary counter of four stages provides the starting point. The normal pattern of a 4-stage binary counter is a repetitive sequence of 16 discrete steps having binary weighting. That is, the first stage output represents the coefficient of 2^0, the second stage output the coefficient of 2^1, the third stage output the coefficient of 2^2, and the fourth stage output the coefficient of 2^3. One can find the decimal equivalent of the counter's outputs by summing the powers of two whose coefficients are "1's." The range of the normal binary counter is therefore from 0 to 15 — sixteen discrete steps.

In a decade counter, six counts must somehow be eliminated. The question of which six to omit is worthy of consideration. If the last six counts are chosen, the remaining ten maintain the same weighting as the normal binary code, namely, 8-4-2-1. This choice is obviously a popular one, and a synchronous version mechanized with DTμL Clocked Flip-Flops and NAND/NOR gates is shown in Fig. 10.

The representation of decimal numbers with four 2-valued symbols, such as the 8-4-2-1 example just given, has the generic name of binary-coded decimal, or BCD. In BCD representations, other weighted codes are possible as well as non-weighted codes. (These two classes are sometimes called fixed weight and variable weight, respectively.) An example of each will be given to demonstrate how the use of other codings can minimize the circuitry involved in a decade counter. These designs, of course, take advantage of the input AND gates in the DTμL Clocked Flip-Flop to achieve their simplicity.

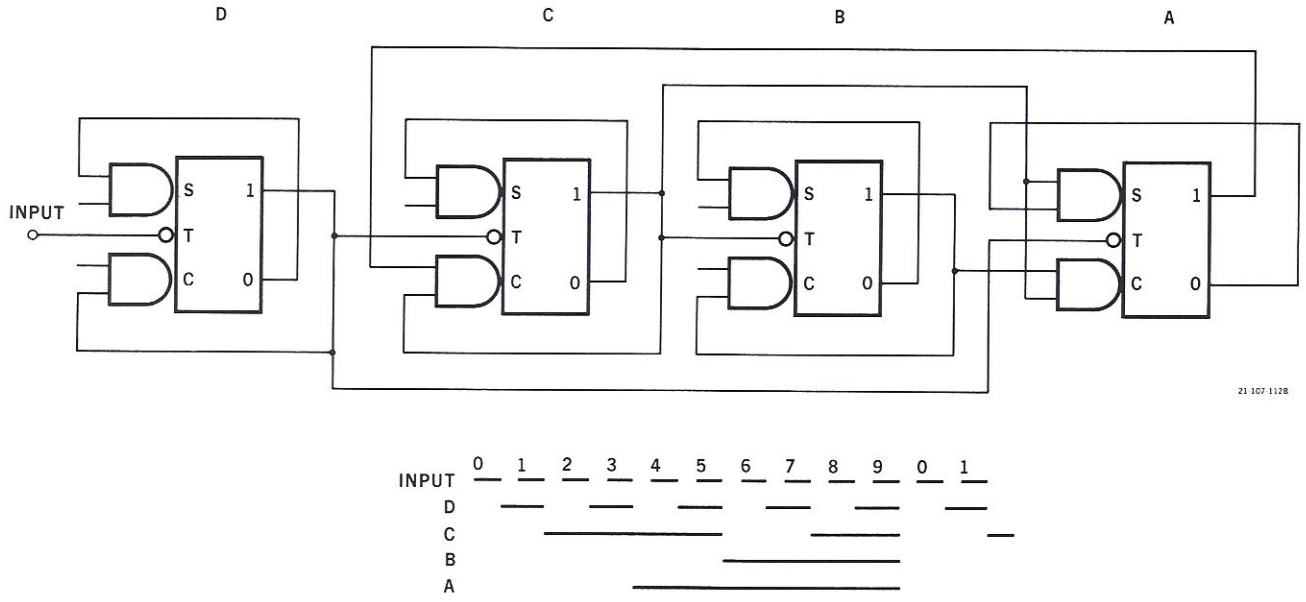


FIG. 11. 4-2-2-1 Ripple-Carry Decade.

Three conditions are required of any weighted decimal code (assuming positive weights only):

- The least significant stage must have the value 1 to permit coding decimal 1.
- The second least significant stage must have a value of either 1 or 2 to permit coding decimal 2.
- The values for the remaining two stages must be chosen so as to permit coding of the remaining decimal numbers.

A consequence of these requirements is that the sum of the assigned values must be equal to or greater than nine.

A weighted code often encountered, particularly in measuring instruments, is the 4-2-2-1 code. A decade counter expressing these weights is given in Fig. 11. Observe that permuting the order of the values does not affect the system in any way, and for graphic convenience this counter is shown as having the weights 1-2-4-2, respectively, from left to right. The remarkable thing about this 4-stage counter using DTμL Clocked Flip-Flop elements is the complete absence of any external gating — a decade counter complete in only four elements!

	SEQUENCE			
COUNT	A	B	C	D
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	1	0	1	0
5	1	0	1	1
6	1	1	0	0
7	1	1	0	1
8	1	1	1	0
9	1	1	1	1

K-MAP

AB \ CD	00	01	11	10
00	0	1	3	2
01	X	X	X	X
11	6	7	9	8
10	X	X	5	4

DECODING

0 = $\bar{B}\bar{C}\bar{D}$ or $\bar{A}\bar{C}\bar{D}$
1 = $\bar{A}\bar{C}D$ or $\bar{B}\bar{C}D$
2 = $\bar{A}C\bar{D}$
3 = $\bar{A}CD$
4 = $A\bar{B}\bar{D}$
5 = $A\bar{B}D$
6 = $A\bar{C}\bar{D}$ or $B\bar{C}\bar{D}$
7 = $B\bar{C}D$ or $A\bar{C}D$
8 = $BC\bar{D}$
9 = BCD

21-107-113A

FIG. 12. Decoding Analysis for 4-2-2-1 Ripple-Carry Decade.

Decoding of the 10 discrete states of this counter requires a total of $3\frac{1}{3}$ triple NAND/NOR gates. Fan-in at each gate is exactly three. The possible alternatives are given in Fig. 12. A Karnaugh map shows the count sequence, and the omitted counts are plotted as "don't cares" to simplify the decoding.

Another weighted code, 8-4-2-1, can also be mechanized with only four DTμL Clocked Flip-Flops (Fig. 13). This is the ripple-carry counterpart of the synchronous counter already given in Fig. 10. Decoding for either of these counters is developed in Fig. 14. Note that 4-input gates are needed to decode counts 0 and 1, while only 2-input gates are needed to decode 8 and 9. Thus, the total number of inputs used is the same as 4-2-2-1 decoding.

An example of a counter having nonweighted coding is shown in Fig. 15. Five DTμL Clocked Flip-Flops are connected as a closed shift register with one crossover.

Since only 10 of the 32 states possible with five flip-flops are actually in use, an additional connection — shown as a dashed line — is included to ensure that the counter does not accidentally enter into a false count sequence (within the range of the 22 unused states). With most integrated circuit families, an additional gate is required to give this protection. The AND gate input of the DTμL Clocked Flip-Flop avoids this necessity.

The advantage of using five flip-flops as a decade is the simplicity of decoding. A Karnaugh map marked with the count sequence and the resulting "don't care"

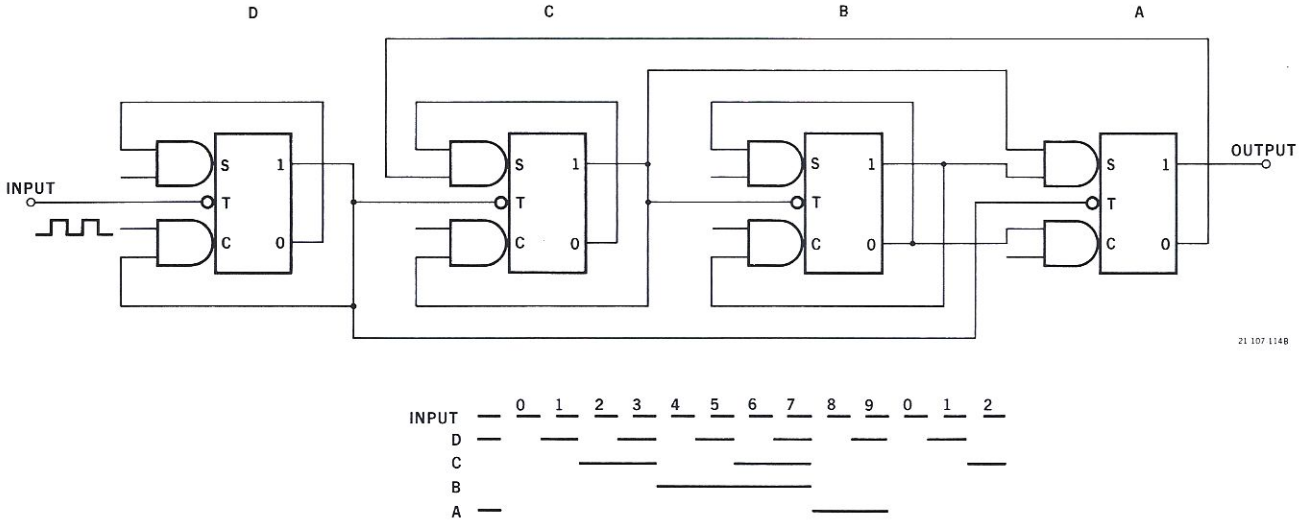


FIG. 13. 8-4-2-1 Ripple-Carry Decade.

	SEQUENCE			
COUNT	A	B	C	D
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
0	0	0	0	0

K-MAP

AB \ CD	00	01	11	10
00	0	1	3	2
01	4	5	7	6
11	X	X	X	X
10	8	9	X	X

DECODING
$0 = \bar{A}\bar{B}\bar{C}\bar{D}$
$1 = \bar{A}\bar{B}\bar{C}D$
$2 = \bar{B}C\bar{D}$
$3 = \bar{B}CD$
$4 = B\bar{C}\bar{D}$
$5 = B\bar{C}D$
$6 = BC\bar{D}$
$7 = BCD$
$8 = A\bar{D}$
$9 = AD$

21-107-115A

FIG. 14. Decoding Analysis for 8-4-2-1 Ripple-Carry Decade.

conditions is given in Fig. 16. Analysis of the K-Map indicates that a fan-in of two is sufficient for a gate decoding any discrete count. With the DTμL Quad 2-input NAND/NOR gate element, only 2-1/2 decoding elements are needed per decade. A more complete discussion of the merits of this counter is given in Fairchild Application Note 85 entitled "Micrologic® Shift Counters."

The coded biquinary counter of Fig. 17 uses the shift counter technique coupled with a ripple-carry to stage four to produce a 4-element decade counter. The feedback line from the stage 2 "1" output to the stage

	SEQUENCE				
COUNT	A	B	C	D	E
0	0	0	0	0	0
1	1	0	0	0	0
2	1	1	0	0	0
3	1	1	1	0	0
4	1	1	1	1	0
5	1	1	1	1	1
6	0	1	1	1	1
7	0	0	1	1	1
8	0	0	0	1	1
9	0	0	0	0	1
0	0	0	0	0	0

K-MAP

ABC \ DE	00	01	11	10
000	0	9	8	X
001	X	X	7	X
011	X	X	6	X
010	X	X	X	X
110	2	X	X	X
111	3	X	5	4
101	X	X	X	X
100	1	X	X	X

DECODING*
$0 = \bar{A}\bar{E}$
$1 = A\bar{B}$
$2 = B\bar{C}$
$3 = C\bar{D}$
$4 = D\bar{E}$
$5 = AE$
$6 = \bar{A}B$
$7 = \bar{B}C$
$8 = \bar{C}D$
$9 = \bar{D}E$

*Sequence Protection Mandatory

21-107-117A

FIG. 16. Decoding Analysis of Shift Counter Decade.

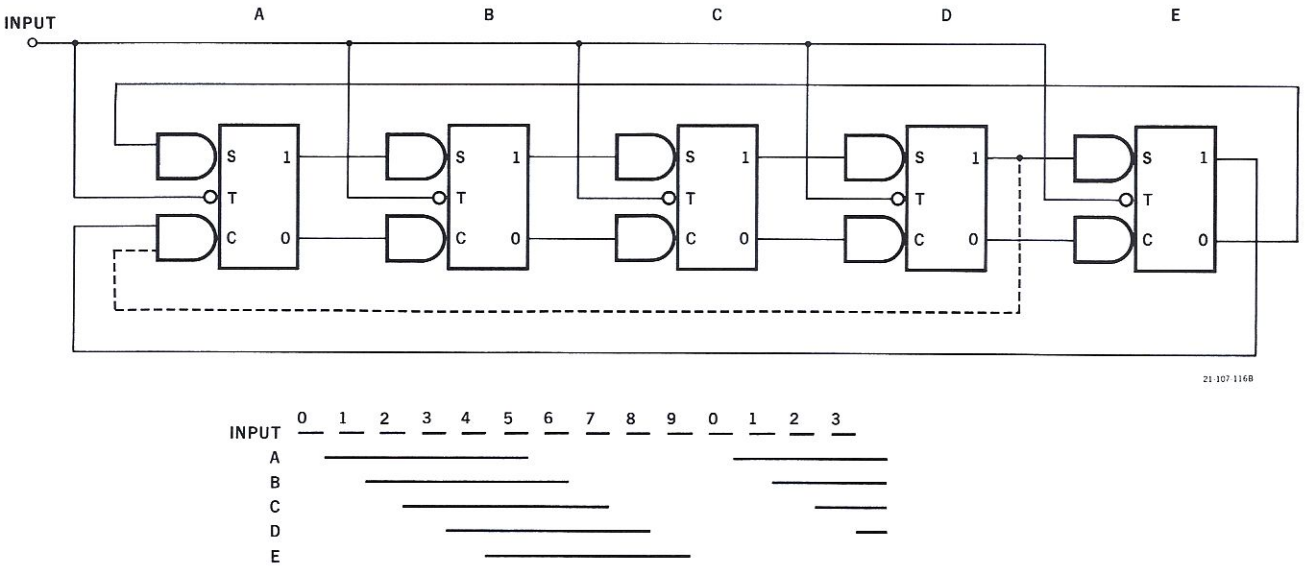


FIG. 15. Decade Shift Counter.

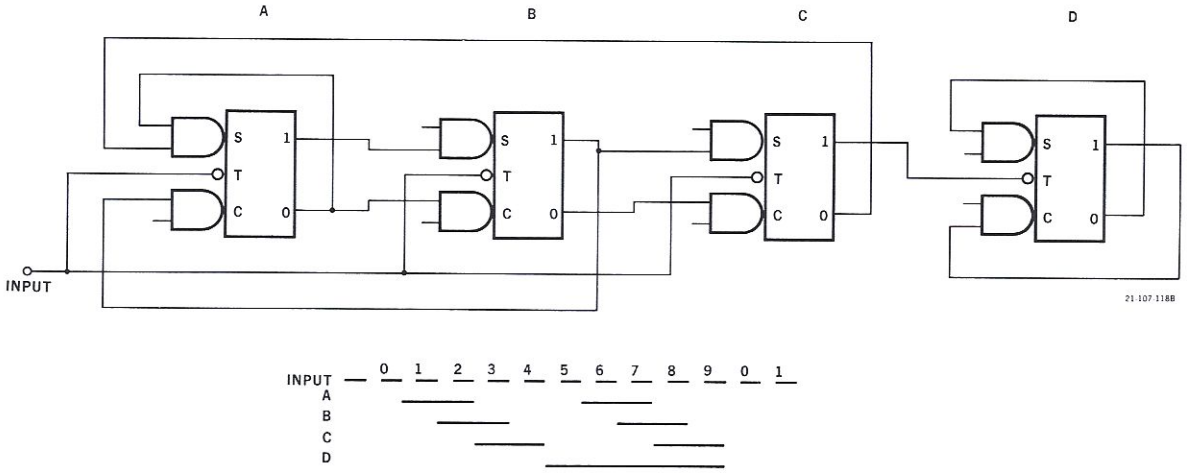


FIG. 17. Coded Biquinary Decade Counter.

1 Clear input reduces the modulus of the first three stages from 6 to 5. Feedback from the stage 1 "0" output to its own Set input resolves input ambiguity at counts 3 and 8. A Karnaugh map of the count sequence (Fig. 18) reveals that 3-input gates are adequate for decoding.

	SEQUENCE			
COUNT	A	B	C	D
0	0	0	0	0
1	1	0	0	0
2	1	1	0	0
3	0	1	1	0
4	0	0	1	0
5	0	0	0	1
6	1	0	0	1
7	1	1	0	1
8	0	1	1	1
9	0	0	1	1
0	0	0	0	0

K-MAP

AB \ CD	00	01	11	10
00	0	5	9	4
01	X	X	8	3
11	2	7	X	X
10	1	6	X	X

DECODING

0 = $\bar{A}\,\bar{C}\,\bar{D}$
1 = $A\,\bar{B}\,\bar{D}$
2 = $B\,\bar{C}\,\bar{D}$ or $A\,B\,\bar{D}$
3 = $\bar{A}\,B\,\bar{D}$ or $B\,C\,\bar{D}$
4 = $\bar{B}\,C\,\bar{D}$
5 = $\bar{A}\,\bar{C}\,D$
6 = $A\,\bar{B}\,D$
7 = $B\,\bar{C}\,D$ or $A\,B\,D$
8 = $\bar{A}\,B\,D$ or $B\,C\,D$
9 = $\bar{B}\,C\,D$

21-107-119A

FIG. 18. Decoding Analysis of Coded Biquinary Decade.

The reversible decade counter of Fig. 19 illustrates the use of the Wired OR function ($\overline{A\bar{B}} \cdot \overline{\bar{C}D} = \overline{AB + CD}$). This particular mechanization saves one-fourth of a Quad Gate per stage over a design in which the Wired OR is not permissible. The control signals labeled Forward and Backward should be mutually exclusive in a positive sense. If both are positive simultaneously, the counter will malfunction; if both are grounded simultaneously, the counter will be disabled.

Binary Adder

The classical binary adder is often divided into two half-adder circuits for reasons of economy. Such a mechanization, while not the fastest approach possible, significantly reduces the amount of logic circuitry involved. The arrangement given in Fig. 20 satisfies all of the functional requirements of a full-adder with only two DTμL 946 elements. The logic delay from summand inputs to sum output is four unit propagation

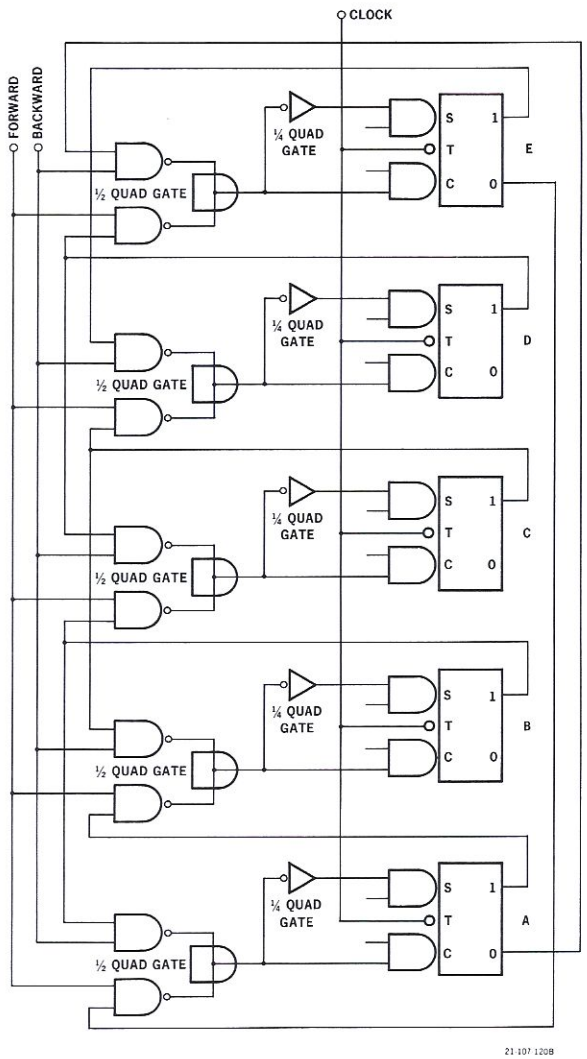


FIG. 19. Reversible MOD-10 Shift Counter.

times. (A statistically sound principle widely accepted among system designers is that the logic delay of chains of four or more cascaded gates may be calculated using typical rather than worst-case propagation delay times.)

The carry scheme illustrated in Fig. 20 is a simple ripple type, but all of the signals needed for a simultaneous carry organization are also present. It is important to note that both the sum and its complement are present. The latter is frequently ignored. Also note that the complement of the carry is neither generated nor needed anywhere as an input.

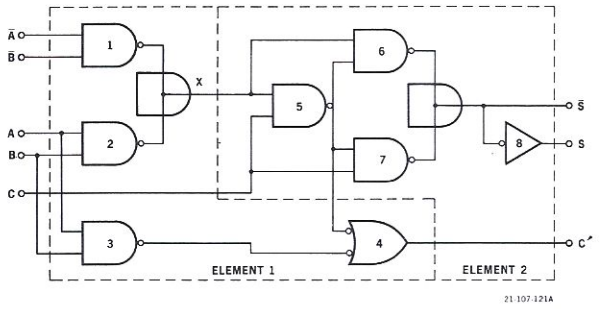


FIG. 20. Binary Full Adder.

The rationale for this mechanization derives from the following factorization of the expressions for the sum and carry. Output carry signals are marked with a prime to distinguish them from the input carry.

$$\begin{aligned} S &= A\bar{B}\bar{C} + \bar{A}B\bar{C} + \bar{A}\bar{B}C + ABC \\ &= (A\bar{B} + \bar{A}B)\bar{C} + (\bar{A}\bar{B} + AB)C \end{aligned}$$

$$\begin{aligned} C' &= A\bar{B}C + \bar{A}BC + AB\bar{C} + ABC \\ &= (A\bar{B} + \bar{A}B)C + AB(\bar{C} + C) \\ &= (A\bar{B} + \bar{A}B)C + AB \end{aligned}$$

Since the following relationship holds,

$$A\bar{B} + \bar{A}B = \overline{\bar{A}\bar{B} + AB}$$

it is apparent from the factored form that the sum is the Exclusive OR of the input carry with the Exclusive OR of the summands A and B. The latter also appear in the expression for the carry.

Two different mechanizations of the Exclusive OR appear in Fig. 20. In the first, composed of NAND gates 1 and 2, both true and complement values of the summands are needed. In the second, composed of NAND gates 5, 6, 7, and inverter 8, complements are not needed.

The Wired OR function is used to advantage in both Exclusive OR circuits. The first Exclusive OR may be analyzed as follows:

$$\text{NAND gate 1} = \overline{\bar{A}\bar{B}}$$

$$\text{NAND gate 2} = \overline{A\bar{B}}$$

As indicated by the AND symbol at the junction, these outputs combine as follows:

$$\overline{\bar{A}\bar{B}} \cdot \overline{A\bar{B}} = \overline{\bar{A}\bar{B} + AB} = \bar{A}B + A\bar{B}$$

The Wired OR is used in the second Exclusive OR circuit as a convenience for providing both true and complement sum outputs.

Gates 3 and 4 create the carry output. NAND gate 3 duplicates NAND gate 2 as the latter is part of a Wired OR function.

The output of NAND gate 5 is the term $(A\bar{B} + \bar{A}B)C$. Both NAND gates 3 and 5 have the proper output polarity for combining in NOR gate 4 to form the carry output with proper polarity for the next cascaded full-adder.